

Logic Programming Engineering – Assignment 2

Dr.-Ing. Sascha Klüppelholz

Exercise 2.1 [Simple cut prediction]

Consider the following program.

```
p(1).  
p(2) :- !.  
p(3).
```

What will be the result of the following queries? Please fill the table by first writing down your predicted answer and then the actual answer when typing the commands into SWI-Prolog. Compare the results and find explanations whenever the outcome differs from what your prediction was.

query	Prologs' answer
p(X).	X = 1; X = 2.
p(X), p(Y).	X = Y, Y = 1; X = 1, Y = 2; X = 2, Y = 1; X = Y, Y = 2.
p(X), !, p(Y).	X = Y, Y = 1; X = 1, Y = 2.

Exercise 2.2 [Cut prediction]

Consider the following Prolog program and **predict** the output to the queries below.

```
top(X,Y) :- predicate(X,Y).  
top(X,X) :- symmetric(X).
```

```
predicate(X,Y) :- cutPoint(1), val4X(X), cutPoint(2), val4Y(Y), cutPoint(3).  
predicate(X,Y) :- symmetric(X), val4Y(Y).
```

```
val4X(a).  
val4X(b).
```

```
val4Y(c).  
val4Y(d).
```

```
symmetric(e).
```

```
cutPoint(_).
```

a) Enumerate all answers to the queries `predicate(X,Y)` and `top(X,Y)`.

<code>predicate(X,Y).</code>	<code>top(X,Y).</code>
<code>X = a, Y = c;</code>	<code>X = a, Y = c;</code>
<code>X = a, Y = d;</code>	<code>X = a, Y = d;</code>
<code>X = b, Y = c;</code>	<code>X = b, Y = c;</code>
<code>X = b, Y = d;</code>	<code>X = b, Y = d;</code>
<code>X = e, Y = c;</code>	<code>X = e, Y = c;</code>
<code>X = e, Y = d.</code>	<code>X = e, Y = d;</code>
	<code>X = Y, Y = e.</code>

b) What changes if `cutPoint(1)` is replaced with a cut?

<code>predicate(X,Y).</code>	<code>top(X,Y).</code>
<code>X = a, Y = c;</code>	<code>X = a, Y = c;</code>
<code>X = a, Y = d;</code>	<code>X = a, Y = d;</code>
<code>X = b, Y = c;</code>	<code>X = b, Y = c;</code>
<code>X = b, Y = d.</code>	<code>X = b, Y = d;</code>
	<code>X = Y, Y = e.</code>

c) What changes if `cutPoint(2)` only is replaced with a cut?

<code>predicate(X,Y).</code>	<code>top(X,Y).</code>
<code>X = a, Y = c;</code>	<code>X = a, Y = c;</code>
<code>X = a, Y = d.</code>	<code>X = a, Y = d;</code>
	<code>X = Y, Y = e.</code>

d) What are all answers to the queries `predicate(X,Y)` and `top(X,Y)` if both `cutPoint(1)` and `cutPoint(2)` are replaced with a cut?

<code>predicate(X,Y).</code>	<code>top(X,Y).</code>
<code>X = a, Y = c;</code>	<code>X = a, Y = c;</code>
<code>X = a, Y = d.</code>	<code>X = a, Y = d;</code>
	<code>X = Y, Y = e.</code>

e) What if only `cutPoint(3)` is replaced with a cut?

<code>predicate(X,Y).</code>	<code>top(X,Y).</code>
<code>X = a, Y = c.</code>	<code>X = a, Y = c;</code>
	<code>X = Y, Y = e.</code>

Exercise 2.3 [Fibonacci numbers]

a) Write a Prolog predicate `fibonacci/2` to compute the n -th Fibonacci number. The Fibonacci sequence is defined as follows: $F_0 = 1$, $F_1 = 2$, and $F_n = F_{n-1} + F_{n-2}$ for $n \geq 2$.

Solution:

```
fibonacci(0,1) :- !.
```

```
fibonacci(1,2) :- !.
```

```

fibonacci(N,X) :-
    N>=2,
    N1 is N-1,
    N2 is N-2,
    fibonacci(N1,FN1),
    fibonacci(N2,FN2),
    X is FN1+FN2.

```

- b) Provide a predicate `fibonacci/4` that computes the n -th Fibonacci number assuming that $F_0 = x_0$ and $F_1 = x_1$, where x_0, x_1 are given as parameters, i.e. $F_0 = x_0$, $F_1 = x_1$, and $F_n = F_{n-1} + F_{n-2}$ for $n \geq 2$. The predicate should have the format: `fibonacci(N,X1,X2,FN)`. Rebase your predicate `fibonacci/2` on `fibonacci/4`.

Solution:

```

fibonacci(0,X0,_,X) :-
    X is X0, !.

```

```

fibonacci(1,_,X1,X) :-
    X is X1, !.

```

```

fibonacci(N,X0,X1,X) :-
    N>=2,
    N1 is N-1,
    N2 is N-2,
    fibonacci(N1,X0,X1,FN1),
    fibonacci(N2,X0,X1,FN2),
    X is FN1+FN2.

```

```

fibonacci(N,X) :- fibonacci(N,1,2,X).

```

- c) Find a solution to the following problem¹: by considering the terms in the Fibonacci sequence whose values do not exceed a certain bound B , what is the sum of the even-valued terms? E.g., if $B = 10$ then, as $F_5 = 13 > B$, only $F_1 = 2$ and $F_4 = 8$ are being summed and the result will be 10. Provide a predicate `fibsum/2` that computes the sum for a given bound.

Solution:

```

fibsum(Bound,Sum) :-
    fibsum(1,2,Bound,Sum).

```

```

fibsum(_,Current,Bound,0) :-
    Current > Bound.

```

```

fibsum(Previous,Current,Bound,Sum) :-
    0 is mod(Current,2),
    Next is Current + Previous,
    fibsum(Current,Next,Bound,SubSum),
    Sum is SubSum + Current, !.

```

¹<https://projecteuler.net/problem=2>

```
fibsum(Previous, Current, Bound, Sum) :-
    Next is Current + Previous,
    fibsum(Current, Next, Bound, Sum).
```

What is the sum if the bound is 4 million?

```
?- fibsum(4000000,X).
X = 4613732.
```

- d) Find a solution to the following problem²: the 10th Fibonacci number $F_{10} = 144$ is the first term to contain three digits. Provide a predicate `fibdigits/2` that returns the first term in the Fibonacci sequence that has $n \in \mathbb{N}$ digits.

Solution:

```
fibdigits(Digits, Solution) :-
    fibdigits(1,2,Digits, Solution).
```

```
fibdigits(_, Current, Digits, Current) :-
    Current > (10**((Digits - 1))), !.
```

```
fibdigits(Previous, Current, Bound, Solution) :-
    Next is Current + Previous,
    fibdigits(Current, Next, Bound, Solution).
```

What is the first term in the Fibonacci sequence to contain 1000 digits?

```
?- fibdigits(1000,X).
```

```
X = 10700662663827589367649805844573968850836838966321516650132352033753145206046940
406218891475824897926578046948881775919574843364666725699595129960304612627480924821
861440694330512347744427502737817530875793916661921492591867595539664228371489431130
746995034395470019854326097230672901928705264472437261177158218255484911205250132014
786129659313817922355596574520395061375514678375432291196021299340482607061753977068
470682028954869026661854351245219003694806413574474709117076197669456910700980243934
396174741037369125032313655321647736970231677550515951735184605799549194109677783732
296657965816465139034881542563101842241902598460880001101862555502454939371136516570
394476295847145485234259504285824253060835444354282126110089928637950480068943303097
732178348645431132057656598684562886168087186938352973506439862976406600007235629179
052070511640776148124918858309459405666883391093509444565763576661516193177537928916
61581327159616877487983821820492520348473874384736771934512787029218636250627816.
```

²<https://projecteuler.net/problem=25>